

A General Form of Attribute Exploration

Daniel Borchmann
 TU Dresden
 Faculty of Mathematics and Sciences
 Institute for Algebra
 daniel.borchmann@mailbox.tu-dresden.de

February 23, 2012

Abstract

We present a general form of attribute exploration, a knowledge completion algorithm from Formal Concept Analysis. The aim of our presentation is not only to extend the applicability of attribute exploration by a general description. It may also allow to view different existing variants of attribute exploration as instances of a general form, which may simplify theoretical considerations.

1 Introduction

Attribute exploration is a well known algorithm within formal concept analysis [9]. Its main application can be summarized as *semi-automatic knowledge base completion*. Within this process, a domain expert is asked about the validity of certain implications in the domain of discourse. Based upon the answer of the domain expert, the algorithm enhances its knowledge until all implications are known to hold or not to hold in the domain, and the algorithm stops.

Attribute exploration has gained much attention since its first formulation, and for certain problems variations of attribute exploration have been devised where the original algorithm was not applicable. Those variations include attribute exploration on partial context [3] and exploration of models of the description logic \mathcal{EL} [1, 2], among others.

However, in almost all variations of attribute exploration that have been devised the overall structure of the algorithm remains the same. Furthermore, all important properties of attribute exploration remain, and one might be tempted to ask whether a general form of attribute exploration can be found that subsumes all many of these variations. The purpose of this work is to present some first considerations into this direction.

We shall proceed as follows. After introducing the mandatory definitions in the first section we briefly revisit the classical description of attribute exploration as it is given in [9]. Starting from this, we motivate our generalizations and summarize the resulting algorithm together with its properties in the succeeding section. We shall have a close look at a special cases which involves *pseudoclosed sets* and results in some very nice results about the

attribute exploration algorithm. Finally, we shall summarize our considerations and give an outlook on further questions.

2 Preliminaries

As attribute exploration is an algorithm from Formal Concept Analysis, we shall begin by introducing some basic definitions from within this field. This includes notions like formal contexts, contextual derivations, implications, partial contexts and pseudoclosed sets. We shall furthermore recall the notion of closure operators on sets, which we need for our considerations.

Let G and M be two sets and let $I \subseteq G \times M$. Then the triple $\mathbb{K} := (G, M, I)$ is called a *formal context*. We shall connect with it the following interpretation: The set G is the set of *objects* of \mathbb{K} , M is the set of *attributes* of \mathbb{K} and (g, m) is an element of the incidence relation I if and only if the object g *has* the attribute m . We may also write gIm if $(g, m) \in I$. If \mathbb{K} is a formal context, then the set of objects, attributes and the incidence relation is denoted by $G_{\mathbb{K}}$, $M_{\mathbb{K}}$ and $I_{\mathbb{K}}$, respectively.

Let us fix a formal context $\mathbb{K} = (G, M, I)$. If $A \subseteq G$, then the set of *common attributes* of A in \mathbb{K} is denoted by

$$A' := \{ m \in M \mid \forall g \in A : gIm \}$$

and likewise for $B \subseteq M$,

$$B' := \{ g \in G \mid \forall m \in B : gIm \}$$

denotes the set of all *common objects* of B in \mathbb{K} . The sets A' and B' are called the (*contextual*) *derivations* of the respective sets, and the operators named $(\cdot)'$ are hence called the *derivation operators* of \mathbb{K} .

Lemma 1 *Let $\mathbb{K} = (G, M, I)$ be a formal context and let $A, A_1, A_2 \subseteq M, B, B_1, B_2 \subseteq G$. Then the following statements hold:*

- i) $A_1 \subseteq A_2 \implies A_2' \subseteq A_1'$
- ii) $B_1 \subseteq B_2 \implies B_2' \subseteq B_1'$
- iii) $A \subseteq A''$
- iv) $B \subseteq B''$
- v) $A' = A'''$
- vi) $B' = B'''$
- vii) $A \subseteq B' \iff A' \supseteq B$

As we view the elements of G as objects with certain attributes from M , we may ask for two sets $A, B \subseteq M$ whether all objects having all attributes from A also have all attributes from B . This can be rewritten in terms of the derivations operators as $A' \subseteq B'$. We shall call the pair (A, B) an *implication on M* and denote it as $A \longrightarrow B$. If \mathbb{K} is a formal context with attribute set M , then we may also say that $A \longrightarrow B$ is an *implication of \mathbb{K}* . Then A is called the *premise* and B the *conclusion* of the implication. If indeed $A' \subseteq B'$, we shall call $A \longrightarrow B$ a *valid implication of \mathbb{K}* , and we may write $\mathbb{K} \models (A \longrightarrow B)$. As $A' \subseteq B' \iff B \subseteq A''$, we can observe that

$$\mathbb{K} \models (A \longrightarrow B) \iff B \subseteq A''.$$

We shall denote with $\text{Imp}(M)$ the set of all implications on M , with $\text{Imp}(\mathbb{K})$ the set of all implications of \mathbb{K} and with $\text{Th}(\mathbb{K})$ the set of all valid implications of \mathbb{K} .

Let $\mathcal{L} \subseteq \text{Imp}(\mathbb{K})$ and let $A \subseteq M$. The set A is *closed under \mathcal{L}* if for all implications $(X \longrightarrow Y) \in \mathcal{L}$ it holds that $X \not\subseteq A$ or $Y \subseteq A$. Let us further define

$$\begin{aligned}\mathcal{L}^0(A) &:= A, \\ \mathcal{L}^1(A) &:= \bigcup \{ Y \mid (X \longrightarrow Y) \in \mathcal{L}, X \subseteq A \}, \\ \mathcal{L}^i(A) &:= \mathcal{L}^1(\mathcal{L}^{i-1}(A)) \quad \text{for } i > 1,\end{aligned}$$

and

$$\mathcal{L}(A) := \bigcup_{i \in \mathbb{N}} \mathcal{L}^i(A).$$

The set $\mathcal{L}(A)$ is then the smallest superset of A that is closed under \mathcal{L} .

The set $\text{Th}(\mathbb{K})$ might be quite large, and to handle this set in practical applications it is desirable to represent it by a small subsets. To see how this is done let $\mathcal{L} \subseteq \text{Imp}(\mathbb{K})$ and let $(A \longrightarrow B) \in \text{Imp}(\mathbb{K})$. Then \mathcal{L} *entails* $A \longrightarrow B$, written as $\mathcal{L} \models (A \longrightarrow B)$, if and only if $B \subseteq \mathcal{L}(A)$. A set $\mathcal{B} \subseteq \text{Imp}(\mathbb{K})$ is called *sound* for \mathcal{L} if every implication from \mathcal{B} is entailed by \mathcal{L} . \mathcal{B} is said to be *complete* for \mathcal{L} if every implication from \mathcal{L} is entailed by \mathcal{B} . If \mathcal{B} is both sound and complete for \mathcal{L} , it is called a *base* for \mathcal{L} . It is called a *non-redundant base* for \mathcal{L} if it is \subseteq -minimal with respect to this property.

Let us denote with $\text{Cn}(\mathcal{L})$ the set of all implications that are entailed by \mathcal{L} . Then

$$\begin{aligned}\mathcal{B} \text{ is sound for } \mathcal{L} &\iff \mathcal{B} \subseteq \text{Cn}(\mathcal{L}), \\ \mathcal{B} \text{ is complete for } \mathcal{L} &\iff \text{Cn}(\mathcal{B}) \supseteq \mathcal{L}.\end{aligned}$$

In particular, \mathcal{B} is a base for \mathcal{L} if and only if $\text{Cn}(\mathcal{B}) = \text{Cn}(\mathcal{L})$.

From all possible bases for \mathcal{L} one can explicitly describe a *canonical base* for \mathcal{L} which has the remarkable property that it has minimal cardinality among all bases for \mathcal{L} . Let $P \subseteq M$. Then P is said to be *pseudoclosed under \mathcal{L}* if

1. $P \neq \mathcal{L}(P)$ and
2. for all pseudoclosed sets $Q \subsetneq P$ it follows $\mathcal{L}(Q) \subseteq P$.

In particular, if $\mathcal{L} = \text{Th}(\mathbb{K})$, then P is said to be a *pseudointent* of \mathbb{K} . Now the *canonical base* for \mathcal{L} is defined as

$$\text{Can}(\mathcal{L}) := \{ P \longrightarrow \mathcal{L}(P) \mid P \text{ pseudoclosed under } \mathcal{L} \}.$$

Formal contexts require a certain kind of complete knowledge about their objects: If $g \in G$ and $m \in M$ then either g has the attribute m or not. Under certain circumstances this might be inappropriate, because it might not be known whether g has the attribute m , or it is simply irrelevant for the task at hand. Therefore we shall introduce *partial contexts*.

Let M be a set. Then a partial context \mathbb{K} is a set of pairs (A, B) with $A, B \subseteq M$ such that $A \cap B = \emptyset$. Such a pair is called a *partial object description* if $A \cup B \neq M$ and a *full object description* if $A \cup B = M$. Intuitively, one can understand partial objects descriptions as a pair of positive attributes, i. e. attributes the corresponding object definitively has, and negative attributes, i. e. attributes the corresponding object definitively does not have. The objects itself are not named in partial contexts.

An *implication* for \mathbb{K} is just an implication on M . Such an implication $(A \longrightarrow B) \in \text{Imp}(M)$ is refuted by \mathbb{K} if there exists a partial object description $(X, Y) \in \mathbb{K}$ such that $A \subseteq X, B \cap Y \neq \emptyset$. If $A \subseteq M$, then the \subseteq -maximal set B such that $A \longrightarrow B$ is not refuted by \mathbb{K} exists and is given by

$$\mathbb{K}(A) := B := M \setminus \bigcup \{ Y \mid (X, Y) \in \mathbb{K}, A \subseteq X \}.$$

As it turns out, the operators $(\cdot)''$, $\mathcal{L}(\cdot)$ and $\mathbb{K}(\cdot)$ are instances of the more abstract notion of *closure operators on sets*. Let again M be a set. Then a function $c: \mathfrak{P}(M) \longrightarrow \mathfrak{P}(M)$ is said to be a *closure operator on M* if and only if

- i) c is *extensive*, i. e. $A \subseteq c(A)$ for all $A \subseteq M$,
- ii) c is *idempotent*, i. e. $c(c(A)) = c(A)$ for all $A \subseteq M$,
- iii) c is *monotone*, i. e. $A \subseteq B \implies c(A) \subseteq c(B)$ for all $A, B \subseteq M$.

Both $(\cdot)''$ and $\mathcal{L}(\cdot)$ are closure operators on their corresponding sets of attributes. A set $A \subseteq M$ is said to be *closed under c* if $c(A) = A$. The set of all closed sets of c , i. e. the image of c , is denoted by $\text{im } c$. A set $P \subseteq M$ is said to be *pseudoclosed under c* if and only if

- i) $P \neq c(P)$ and
- ii) for all pseudoclosed $Q \subseteq P$, it holds that $c(Q) \subseteq P$.

We shall write $c_1(\cdot) \subseteq c_2(\cdot)$ for two closure operators c_1, c_2 on a set M if and only if $c_1(A) \subseteq c_2(A)$ for all $A \subseteq M$.

3 Classical Attribute Exploration

Given a finite set M , attribute exploration semi-automatically tries to determine the set of implications that are valid in a certain domain. Together with a set \mathcal{K} of already known

valid implications and a formal context \mathbb{K} of valid examples, attribute exploration generates implications $A \longrightarrow B$ that hold in \mathbb{K} but are not entailed by \mathcal{K} . Those implications are asked to the expert for validity. If $A \longrightarrow B$ holds in the domain of discourse, it is added to the set \mathcal{K} . Otherwise the expert has to present a counterexample for $A \longrightarrow B$ that is added to the formal context \mathbb{K} . The procedure terminates if there are no such implications left.

To describe attribute exploration more formally, let us define what is meant by a domain expert.

Definition 2 Let M be a set. A *domain expert* on M is a function

$$p: \text{Imp}(M) \longrightarrow \{\top\} \cup \mathfrak{P}(M),$$

where \top is a special symbol not equal to any subset of M , such that the following conditions hold

- i) If $X \longrightarrow Y$ is an implication on M such that $p(X \longrightarrow Y) = C \neq \top$, then $X \subseteq C, Y \not\subseteq C$. (*p gives counterexamples for false implications*)
- ii) If $A \longrightarrow B$ and $X \longrightarrow Y$ are implications on M such that $p(A \longrightarrow B) = \top$ and $p(X \longrightarrow Y) = C \neq \top$, then C is closed under $\{A \longrightarrow B\}$, i. e. $A \not\subseteq C$ or $B \subseteq C$. (*counterexamples do not invalidate correct implications*)

If $p(A \longrightarrow B) = \top$, then we say that p *confirms* $A \longrightarrow B$. Otherwise we say that p *rejects* the implication and we call the set $C = p(A \longrightarrow B) \neq \top$ a *counterexample from p for $A \longrightarrow B$* . Finally, the *theory* of p is just the set of implications that p confirms, i. e.

$$\text{Th}(p) := p^{-1}(\{\top\}) = \{A \longrightarrow B \mid p(A \longrightarrow B) = \top\}. \quad \diamond$$

An immediate consequence of the definition is the following observation.

Lemma 3 Let \mathcal{L} be a set of implications such that a given domain expert p confirms every implication in \mathcal{L} . If $\mathcal{L} \models (A \longrightarrow B)$, then p confirms $A \longrightarrow B$ as well.

Proof Suppose that $p(A \longrightarrow B) = C \neq \top$. Then C is closed under \mathcal{L} . This means that $\mathcal{L}(C) = C$. Since $\mathcal{L} \models (A \longrightarrow B)$, from $A \subseteq C$ it follows that

$$B \subseteq \mathcal{L}(A) \subseteq \mathcal{L}(C) = C,$$

i. e. C is not a counterexample for $A \longrightarrow B$, a contradiction. \square

Before we are able to describe the attribute exploration algorithm more formally, we need to give another definition.

Definition 4 Let M be a finite set and let $<$ be a total order on M . Then for $A, B \subseteq M$ and $i \in M$ we define

$$A \prec_i B : \Longleftrightarrow i = \min_{<}(A \Delta B),$$

where $A \Delta B = (A \setminus B) \cup (B \setminus A)$ is the symmetric difference between A and B . If $A \prec_i B$, we say that A is *lectically smaller than B at i* . Furthermore, A is *lectically smaller than B* , written as $A \prec B$, if there exists $i \in M$ such that $A \prec_i B$. Finally,

$$A \preceq B \iff A = B \text{ or } A \prec B. \quad \diamond$$

It is easy to see that \preceq constitutes a linear ordering on $\mathfrak{P}(M)$.

We are now able to describe the process of attribute exploration in a more formal way.

Algorithm 5 (Classical Attribute Exploration) Let M be a finite set, \mathbb{K} be a formal context with attribute set M and let $\mathbb{K} \subseteq \text{Imp}(M)$ and let p be a domain expert on M . Suppose that $\mathcal{K} \subseteq \text{Th}(p) \subseteq \text{Th}(\mathbb{K})$.

- i) Initialize P to the lectically first closed set of $\mathcal{K}(\cdot)$.
- ii) If $P'' = P$, then go to v. Otherwise let $r := (P \longrightarrow P'')$.
- iii) If p confirms r , then add r to \mathcal{K} .
- iv) If p gives a counterexample C for r , add a new object to \mathbb{K} which has exactly the attributes in C .
- v) Let Q be the lectically next closed set after P of \mathcal{K} . If there is none left, terminate. Otherwise, set P to Q and go to ii.

In any iteration, the current value of \mathcal{K} is called the set of *currently known implications* and the current value of \mathbb{K} is called the *current working context*. \diamond

A first easy observation for this algorithm is the following: Suppose the expert p is called with an implication $A \longrightarrow B$ during the run of the algorithm. Let \mathcal{K} be the currently known implications at this time, and let likewise \mathbb{K} denote the current working context. Then for each $m \in B$ both $\text{Th}(p) \models (P \longrightarrow \{m\})$ and $\text{Th}(p) \not\models (P \longrightarrow \{m\})$ is possible. In other words, the question whether $\text{Th}(p) \models (P \longrightarrow \{m\})$ is not influenced by the values of \mathcal{K} and \mathbb{K} but solely depends on how the expert p answers. Hence all questions to the expert can be seen as *non-redundant*.

This property is very important especially in the presence of human experts which may not only be expensive to answer but might also get impatient when getting asked implications the algorithm could have inferred by itself. Therefore, this property should of course also hold for our generalized formulation of the attribute exploration, and it does, as we shall see.

But before we do so, we shall mark down some of the major properties of this attribute exploration algorithm.

Theorem 6 Let M be a finite set, $<$ a total order on M , \mathbb{K} a formal context with attribute set M , \mathcal{K} a set of implications on M and let p be a domain expert on M , such that p confirms \mathcal{K} and all implications confirmed by p hold in \mathbb{K} , i. e. $\mathcal{K} \subseteq \text{Th}(p) \subseteq \text{Th}(\mathbb{K})$.

- i) The attribute exploration algorithm terminates with \mathbb{K} , \mathcal{K} and p as input.

- ii) Let \mathcal{K}' and \mathbb{K}' be the values corresponding to \mathcal{K} and \mathbb{K} after the last iteration of the attribute exploration algorithm. Then \mathcal{K}' is a base for $\text{Th}(\mathbb{K}')$.
- iii) $\text{Th}(p) = \text{Th}(\mathbb{K}')$ and the corresponding closure operator coincides with $\mathcal{K}'(\cdot)$.
- iv) The cardinality of $\mathcal{K}' \setminus \mathcal{K}$ is the smallest possible.
- v) The premises in $\mathcal{K}' \setminus \mathcal{K}$ are the \mathcal{K} -pseudoclosed of $\text{Th}(\mathbb{K}')$. Thereby, a set $P \subseteq M$ is said to be \mathcal{K} -pseudoclosed under \mathcal{L} for $\mathcal{K}, \mathcal{L} \subseteq \text{Imp}(M)$, if and only if
 - i) $P = \mathcal{K}(P)$,
 - ii) $P \neq \mathcal{L}(P)$,
 - iii) for each \mathcal{K} -pseudoclosed set $Q \subsetneq P$ of \mathcal{L} it holds that $\mathcal{L}(Q) \subseteq P$.

All but the last statement of the theorem are known from [9, 11, 6]. The last statement has been mentioned partially in [11] and has been proven completely in [5].

4 Generalizing Attribute Exploration

We shall now proceed by investigating the above description of attribute exploration for possible generalizations. While doing so, we shall not only generalize certain aspect of the algorithm but also generalize those aspects intuitively. The main aim of our generalization is to describe attribute exploration in more abstract terms, to allow applications of the algorithm beyond those of the classical algorithm.

Let p be a domain expert on a set M . We start with an informal introduction of our generalizations, of which we shall name three:

1. The use of the initial formal context \mathbb{K} and the background knowledge \mathcal{K} can be reduced to their corresponding closure operators $(\cdot)''$ and $\mathcal{K}(\cdot)$. The only major problem here is the handling of counterexamples, which we shall discuss latter in detail. Hence instead of passing the attribute exploration algorithm a formal context and some background knowledge in the form of a set of valid implications, we instead provide two closure operators c_{univ} and c_{cert} on the set M .

The closure operator c_{univ} takes the place of $\text{Th}(\mathbb{K})(\cdot)$ and represents the *universal knowledge* we already have about our domain of discourse. If $A \subseteq M$ is a set of attributes, then $c_{\text{univ}}(A)$ represents the attributes that *can* follow from A . Seen from another perspective, $M \setminus c_{\text{univ}}(A)$ is the set of attributes that *do not follow* from A .

In contrast to this, the closure operator c_{cert} represents the *certain knowledge* we already have. In other words, $c_{\text{cert}}(A)$ is the set of all attributes that *definitively follow* from A . This closure operators hence takes the place of the set \mathcal{K} of initially known implications.

Clearly, we need to have $c_{\text{cert}}(\cdot) \subseteq \text{Th}(p)(\cdot) \subseteq c_{\text{univ}}(\cdot)$.

2. When providing counterexamples, we observe that we actually do not need to *completely specify* them. It merely is sufficient to provide information on which attributes a certain object has and which it not, as long as this information contradicts a proposed implication. We shall take this approach and extend the algorithm to store those counterexamples in a partial context. This idea has also been discussed in [3, 8].
3. The implications which are proposed to the expert are of a very special form, which guarantees certain optimality statements about the algorithm. However, for the main application of knowledge acquisition and knowledge completion, this rather special form can be viewed as a certain kind of optimization. To drop it, we may rather say that in any iteration step of the attribute exploration algorithm, we search for an *undecided implication* with respect to the current values of c_{cert} and c_{univ} , i. e. an implication $A \longrightarrow B$ on M such that $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{univ}}(A)$ and where both A and B are finite. For such an implication we cannot infer from c_{cert} and c_{univ} whether attributes $c_{\text{univ}}(A) \setminus B$ follow from A or not, and hence we have to ask the domain expert.

We shall take these observations as guidelines for our further considerations. We start by generalizing our notion of a domain expert such that we allow partial counter examples. Next we present and discuss our general form of attribute exploration that incorporates the above mentioned ideas. For this we shall also prove correctness and non-redundancy of the questions asked to the expert. Subsequently, we shall have a closer look on how to compute undecided implications in our general setting as it is done in the classical case.

Definition 7 Let M be a set. A function $q: \text{Imp}(M) \longrightarrow \{\top\} \cup \mathfrak{P}(M)^2$ is said to be a *partial domain expert on M* if and only if \top is an element not in $\mathfrak{P}(M)^2$ and the following conditions hold:

1. If for $(A \longrightarrow B) \in \text{Imp}(M)$ it holds that $q(A \longrightarrow B) = (C, D) \neq \top$, then $C \cap D = \emptyset$, $A \subseteq C$ and $B \cap D \neq \emptyset$. (*q gives sufficient counterexamples for false implications*)
2. If $(A \longrightarrow B), (X \longrightarrow Y) \in \text{Imp}(M)$ are such that $q(A \longrightarrow B) = \top$ and $q(X \longrightarrow Y) = (C, D) \neq \top$, then if $A \subseteq C$ then $B \cap D = \emptyset$. (*counterexamples do not refute correct implications*)

As in the case for domain experts, we say that q *confirms* an implication $A \longrightarrow B$ if and only if $q(A \longrightarrow B) = \top$. Otherwise we say that q *rejects* the implication and we call $q(A \longrightarrow B) \neq \top$ a *counterexample from q* for $A \longrightarrow B$. $\text{Th}(q)$ shall denote the set of all implications on M that are confirmed by q . \diamond

The counterexamples given by a partial domain expert can be seen as partial object descriptions that are enough to invalidate a given implication.

Let us first investigate immediate consequences from the definition. One of those is the fact, as one would expect, that $\text{Th}(q)$ is closed under entailment, i. e. $\text{Cn}(\text{Th}(q)) = \text{Th}(q)$.

Lemma 8 Let $\mathcal{L} \subseteq \text{Imp}(M)$ for a set M and let q be a partial domain expert on M , such that q confirms all implications in \mathcal{L} . If $\mathcal{L} \models (A \longrightarrow B)$ for some $(A \longrightarrow B) \in \text{Imp}(M)$, then q confirms $A \longrightarrow B$ as well.

Proof Suppose that $q(A \rightarrow B) = (C, D)$ is a counterexample from q for $A \rightarrow B$. Then $A \subseteq C$. Now $\mathcal{L}(C) \subseteq M \setminus D$ by the second condition on partial domain experts. Since $\mathcal{L} \models (A \rightarrow B)$, it follows that $B \subseteq \mathcal{L}(A) \subseteq \mathcal{L}(C) \subseteq M \setminus D$. Therefore, $B \cap D = \emptyset$, contradicting the fact that (C, D) is a counterexample for $A \rightarrow B$ from q . \square

Lemma 9 *If (C, D) is a counterexample given by a partial domain expert q on M , then $\text{Th}(q)(C) \cap D = \emptyset$.*

Proof By Lemma 8, q confirms $C \rightarrow \text{Th}(q)(C)$. Therefore, by the second condition in the definition of q , it follows $D \cap \text{Th}(q)(C) = \emptyset$, as required. \square

Lemma 10 *For a partial context \mathbb{K} with attribute set M and a partial domain expert q on M it holds that $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ if and only if $\text{Th}(q)(C) \subseteq M \setminus D$ for each $(C, D) \in \mathbb{K}$.*

Proof $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ implies $\text{Th}(q)(C) \cap D = \emptyset$ for each $(C, D) \in \mathbb{K}$, which is equivalent to $\text{Th}(q)(C) \subseteq M \setminus D$.

For the converse let $\text{Th}(q)(C) \cap D = \emptyset$ for all $(C, D) \in \mathbb{K}$. Let $A \subseteq M$. Then for every $(C, D) \in \mathbb{K}$ with $A \subseteq C$, it follows that $\text{Th}(q)(A) \cap D \subseteq \text{Th}(q)(C) \cap D = \emptyset$. Therefore

$$\text{Th}(q)(A) \cap \bigcup \{ D \mid (C, D) \in \mathbb{K}, A \subseteq C \} = \emptyset$$

and hence $\text{Th}(q)(A) \subseteq \mathbb{K}(A)$ as required. \square

With those observations at hand, we are now able to state our generalized formulation of the attribute exploration algorithm.

Algorithm 11 (General Attribute Exploration) Let M be a set, $c_{\text{cert}}, c_{\text{univ}}$ closure operators on M and q a partial domain expert M , such that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$.

- i. Let $\mathbb{K} = \emptyset$.
- ii. Let $A \subseteq M$ be finite and such that there exists a finite set $B \subseteq M$ with $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{univ}}(A)$. If there is no such set, terminate with output \mathbb{K} and c_{cert} . Otherwise consider the implication $A \rightarrow B$.
- iii. If q confirms $A \rightarrow B$, then update c_{cert} to be the closure operators whose closed sets are exactly the closed sets of c_{cert} that are also closed under $\{ A \rightarrow B \}$.
- iv. Otherwise let $(C, D) = q(A \rightarrow B)$ be a counterexample from q for $A \rightarrow B$. Add (C, D) to \mathbb{K} .
- v. Replace all counterexamples $(C, D) \in \mathbb{K}$ by (C', D') , where

$$\begin{aligned} C' &:= c_{\text{cert}}(C), \\ D' &:= D \cup \{ m \in M \setminus D \mid c_{\text{cert}}(C \cup \{ m \}) \cap D \neq \emptyset \}. \end{aligned}$$

- vi. Update c_{univ} to be the closure operator given by

$$X \mapsto c_{\text{univ}}(X) \cap \mathbb{K}(X)$$

for all $X \subseteq M$.

vii. Go to ii. ◇

Starting from this reformulation of the attribute exploration algorithm we shall now consider the properties this algorithm has. We shall show in this section that the algorithm, as in the classical case, does not ask questions it could infer itself. Furthermore, the algorithm is correct in the sense that it returns a complete description of the domain the given partial domain expert represents. Termination, however, cannot be shown in general, and we shall only give some sufficient condition.

The results in the minimality of the resulting set of confirmed implications does not hold in this general setting. For this, we have to generate the implications asked to the expert in a way similar to the classical case. We shall discuss this in more detail in the next section.

To discuss the properties of Algorithm 11, we need the following result.

Lemma 12 *At the end of every iteration of the generalized attribute exploration algorithm it holds that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$ for the current values of c_{cert} and c_{univ} . In particular, $c_{\text{cert}}(X) \subseteq \mathbb{K}(X)$ holds for all $X \subseteq M$ at the end of every iteration.*

Proof We prove the claim by induction. For the base case we observe that $\mathbb{K} = \emptyset$ and therefore $\mathbb{K}(X) = M$ for all $X \subseteq M$. Furthermore $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$ by the prerequisites of the algorithm.

For the induction step assume that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$ holds at the beginning of the current iteration. Assume $A, B \subseteq M$ finite such that $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{univ}}(A)$, for otherwise nothing has to be shown. We now distinguish two cases:

- i. q confirms $A \longrightarrow B$. Then c_{cert} is updated to the value of

$$c'_{\text{cert}} = X \mapsto c_{\text{cert}}(\mathcal{L}(c_{\text{cert}}(X)))$$

where $\mathcal{L} = \{A \longrightarrow B\}$ and $X \subseteq M$. Since q confirms $A \longrightarrow B$ and $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot)$, it follows that $c'_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot)$.

In the situation before step v, by Lemma 9 for every element $(C, D) \in \mathbb{K}$ it holds that $\text{Th}(q)(C) \cap D = \emptyset$ and hence $c'_{\text{cert}}(C) \cap D = \emptyset$. Moreover, $C' := c'_{\text{cert}}(C)$ is also disjoint to

$$D' := D \cup \{m \in M \setminus D \mid c'_{\text{cert}}(C \cup \{m\}) \cap D \neq \emptyset\}$$

and $(C' \longrightarrow \{m\}) \notin \text{Th}(q)$ for $m \in D' \setminus D$. Therefore, after step v, $\text{Th}(q)(C') \subseteq M \setminus D'$ for every $(C', D') \in \mathbb{K}$. Then by Lemma 10, $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ and therefore $c'_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot) \cap \mathbb{K}(\cdot)$ as required.

- ii. q gives (X, Y) as a counterexample for $A \longrightarrow B$. Then in this iteration the value of c_{cert} is not changed. The counterexample that is effectively added to \mathbb{K} is then

$$(X', Y') = (c_{\text{cert}}(X), Y \cup \{m \in M \setminus Y \mid c_{\text{cert}}(X \cup \{m\}) \cap Y \neq \emptyset\}).$$

Since $\text{Th}(q)(X') \subseteq M \setminus Y'$, from Lemma 10 and the induction hypothesis it follows that $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$. Together with $\text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$ we obtain $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot) \cap \mathbb{K}(\cdot)$ as required. □

We shall at first investigate the already mentioned property that questions asked to the expert are somehow non-redundant. We state this kind of non-redundancy as the fact that the answer to a proposed implication is not predetermined by the current knowledge or by the answers given so far.

Theorem 13 *Let M be a set, $c_{\text{cert}}, c_{\text{univ}}$ closure operators on M and q a partial domain expert on M such that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$. Suppose that we are in the $n + 1$ iteration of Algorithm 11 and suppose that the implication $A \longrightarrow B$ is asked to the expert q .*

Then for each $m \in B$ there exist two partial domain experts q_1, q_2 which return the same values as q in all iterations $i \in \{1, \dots, n\}$ and satisfy $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q_1)(\cdot), \text{Th}(q_2)(\cdot) \subseteq c_{\text{univ}}(\cdot)$, such that q_1 rejects $A \longrightarrow \{m\}$ and q_2 confirms $A \longrightarrow \{m\}$.

Proof Let $c_{\text{cert}}^i, c_{\text{univ}}^i, \mathbb{K}^i$ be the values of the corresponding closure operators and the current working context in iteration $i \in \{1, \dots, n\}$, respectively. Furthermore, let $A_i \longrightarrow B_i$ be the implication asked in iteration i . Finally, let \top be a symbol not equal to any subset of M .

We then define q_1 as follows:

$$q_1(A \longrightarrow B) = \begin{cases} q(A \longrightarrow B) & \text{if } (A \longrightarrow B) = (A_i \longrightarrow B_i) \text{ for some } i, \\ \top & \text{if } B \subseteq c_{\text{cert}}^n(A), \\ (c_{\text{cert}}(A), M \setminus c_{\text{cert}}(A)) & \text{otherwise,} \end{cases}$$

for all $(A \longrightarrow B) \in \text{Imp}(M)$. Then q_1 is a partial domain expert on M and $\text{Th}(q_1) = \text{Th}(c_{\text{cert}}^n)$. Since $c_{\text{cert}}^n(\cdot) \subseteq c_{\text{univ}}(\cdot)$ by Lemma 12 and $m \notin c_{\text{cert}}^n(A)$, q_1 rejects $A \longrightarrow \{m\}$.

To construct q_2 we consider the formal context \mathbb{K} with object set \mathbb{K}^n , attribute set M and incidence relation $I_{\mathbb{K}}$ given by

$$(C, D) I_{\mathbb{K}} x \iff \begin{cases} x \in c_{\text{cert}}^n(C \cup \{m\}) & \text{if } m \notin D \\ x \in C & \text{otherwise} \end{cases}$$

for all $(C, D) \in \mathbb{K}^n$ and $x \in M$. By step v in Algorithm 11, all object intents of \mathbb{K} are closed under c_{cert}^n , therefore $\text{Th}(c_{\text{cert}}^n) \subseteq \text{Th}(\mathbb{K})$. Together with $c_{\text{cert}}(\cdot) \subseteq c_{\text{cert}}^n(\cdot)$ follows $c_{\text{cert}}(\cdot) \subseteq \text{Th}(\mathbb{K})(\cdot)$.

We now define q_2 by

$$q_2(A \longrightarrow B) = \begin{cases} q(A \longrightarrow B) & \text{if } (A \longrightarrow B) = (A_i \longrightarrow B_i) \text{ for some } i, \\ \top & \text{if } B \subseteq A'' \cap c_{\text{univ}}^n(A), \\ (X, M \setminus X) & \text{with } X = A'' \cap c_{\text{univ}}^n(A) \text{ otherwise} \end{cases}$$

for all $(A \longrightarrow B) \in \text{Imp}(M)$. Then q_2 is a partial domain expert with $\text{Th}(q_2) = \text{Th}(\mathbb{K}) \cap \text{Th}(c_{\text{univ}}^n)$. For this we observe that for $(C, D) \in \mathbb{K}^n$, if $m \notin D$, then $c_{\text{cert}}^n(C \cup \{m\}) \cap D = \emptyset$ by step v. Therefore, the counterexamples given for some implication $A_i \longrightarrow B_i$ from q can also be given by q_2 .

Since $c_{\text{cert}}(\cdot) \subseteq \text{Th}(\mathbb{K})(\cdot)$ and $c_{\text{cert}}(\cdot) \subseteq c_{\text{univ}}^n(\cdot)$, it follows that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q_2)(\cdot) \subseteq c_{\text{univ}}(\cdot)$.

Furthermore, $m \in c_{\text{univ}}''(A)$ and since \mathbb{K}^n does not reject $A \rightarrow B$, it follows that for each $(C, D) \in \mathbb{K}^n$ with $A \subseteq C$ that $m \notin D$. Hence, $m \in A''$ and therefore q_2 confirms $A \rightarrow B$ as required. \square

One of the crucial features of attribute exploration is that it returns a complete description of the domain of discourse upon termination. This property does also hold for our generalized formulation.

Theorem 14 *Let M be a set, $c_{\text{cert}}, c_{\text{univ}}$ closure operators on M and let q be a partial domain expert on M . Furthermore, suppose that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$.*

Suppose that Algorithm 11 terminates on input $c_{\text{cert}}, c_{\text{univ}}$ and q and denote the returned partial context by \mathbb{K} and the returned closure operator by c . Let $X \subseteq M$ such that $c(X)$ is finite.

- i. $\text{Th}(q)(X) = c(X)$.
- ii. $c(X) = c_{\text{univ}}(X) \cap \mathbb{K}(X)$.
- iii. *Let \mathcal{K} be the set of all implications which have been confirmed by q during the run of the algorithm. Define $c'(X)$ to be the smallest set that contains X and is closed under both c_{cert} and $\mathcal{K}(\cdot)$. Then $c'(X) = c(X)$.*
- iv. *Let $\bar{\mathbb{K}} = (\mathbb{K}, M, I)$ where*

$$(C, D) \text{Im} \iff m \in C.$$

Then

$$c(X) = c_{\text{univ}}(X) \cap X'',$$

where $(\cdot)''$ denotes the double derivation operator in $\bar{\mathbb{K}}$.

Proof By Lemma 12, $c'_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c'_{\text{univ}}(\cdot)$ holds at the end of every iteration in the run of the algorithm, where c'_{cert} and c'_{univ} denote the current values of the corresponding closure operators. Since the algorithm terminates, $c'_{\text{cert}}(Y) = c'_{\text{univ}}(Y)$ holds in the last iteration for all $Y \subseteq M$ if $c'_{\text{cert}}(Y)$ is finite. Since $c = c'_{\text{cert}}$ and $c'_{\text{cert}}(X) \subseteq \text{Th}(q)(X) \subseteq c'_{\text{univ}}(X)$, the first assertion follows.

By induction on the number of iterations of the algorithm, one can see that at the end of every iteration of the algorithm it holds that $c'_{\text{univ}}(X) = c_{\text{univ}}(X) \cap \mathbb{K}(X)$, where c'_{univ} is the current value of the upper closure operator, c_{univ} is the original value of the upper closure operator and \mathbb{K} is the current working context. Since the algorithm terminates, $c'_{\text{univ}}(X) = c(X)$ holds in the last iteration and the second claim follows.

Suppose that the algorithm is in a certain iteration and suppose that \mathcal{K}' is the set of confirmed implications up to now. By induction we see that if c'_{cert} is the current value of the lower closure operator, then $c'_{\text{cert}}(X)$ is the smallest set containing X that is closed both under c_{cert} and $\mathcal{K}'(\cdot)$. As c is the last value of the lower closure operator during the run of the algorithm, $c(X) = c'(X)$, which shows the third claim.

For the last claim we observe the following relations:

$$\begin{aligned} X'' &= \bigcap_{(C,D) \in \mathbb{K}, X \subseteq C} C \\ &\subseteq \bigcap_{(C,D) \in \mathbb{K}, X \subseteq C} M \setminus D \\ &= \mathbb{K}(X). \end{aligned}$$

By step v of the algorithm, C is closed under c for every $(C, D) \in \mathbb{K}$. Therefore, $c(X) \subseteq X''$. Together this yields

$$c_{\text{univ}}(X) \cap c(X) \subseteq c_{\text{univ}}(X) \cap X'' \subseteq c_{\text{univ}}(X) \cap \mathbb{K}(X)$$

and since $c(X) \subseteq c_{\text{univ}}(X)$ and $c(X) = c_{\text{univ}}(X) \cap \mathbb{K}(X)$, the last claim follows. \square

Termination of the generalized attribute exploration algorithm is not guaranteed in general (i. e. when M is infinite and c_{cert} and c_{univ} are arbitrary). Hence, termination normally has to be shown for the concrete application at hand. We can, however, give some sufficient condition which may still be helpful.

Theorem 15 *The general attribute exploration algorithm with input c_{cert} , c_{univ} and a partial domain expert q terminates if there are only finitely many closure operators c on M such that $c_{\text{cert}}(\cdot) \subsetneq c(\cdot) \subsetneq c_{\text{univ}}(\cdot)$.*

Proof The claim follows easily if we can show that in every iteration of attribute exploration either the value of c_{cert} is updated to a new value c'_{cert} such that $c_{\text{cert}} \subsetneq c'_{\text{cert}} \subseteq c_{\text{univ}}$ or, likewise, if the value for c_{univ} is updated to a new value c'_{univ} such that $c_{\text{cert}} \subseteq c'_{\text{univ}} \subsetneq c_{\text{univ}}$.

Let A be such that $c_{\text{cert}}(A) \neq c_{\text{univ}}(A)$ and let $B \subseteq M$ be finite such that $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{univ}}(A)$. If q confirms $A \rightarrow B$, then c_{cert} is updated to the value

$$c'_{\text{cert}}(X) = c_{\text{cert}}(\mathcal{L}(c_{\text{cert}}(X))),$$

where $\mathcal{L} = \{A \rightarrow B\}$ and $X \subseteq M$. Clearly, $c_{\text{cert}}(\cdot) \subsetneq c'_{\text{cert}}(\cdot)$ and by Lemma 12, $c'_{\text{cert}}(\cdot) \subseteq c_{\text{univ}}(\cdot)$.

If q yields a counterexample (C, D) for $A \rightarrow B$, then the new value c'_{univ} for c_{univ} is computed by

$$c'_{\text{univ}}(X) = c_{\text{univ}}(X) \cap \mathbb{K}(X)$$

for $X \subseteq M$. It follows that $c'_{\text{univ}}(\cdot) \subseteq c_{\text{univ}}(\cdot)$ and $c'_{\text{univ}}(A) \subseteq c_{\text{univ}}(A) \setminus D \subsetneq c_{\text{univ}}(A)$, since $C \subseteq A$, $B \subseteq c_{\text{univ}}(A)$ and $B \cap D \neq \emptyset$. By Lemma 12 it follows that $c_{\text{cert}}(X) \subseteq \mathbb{K}(X)$ for all $X \subseteq M$. Hence $c_{\text{cert}}(\cdot) \subseteq c'_{\text{univ}}(\cdot) \subsetneq c_{\text{univ}}(\cdot)$ as required. \square

Of course, if after finitely many iterations the situation of the theorem is reached, the generalized attribute exploration will terminate as well.

5 Computing Undecided Implications

We have seen that a lot of the useful properties of attribute exploration remain true in our generalized form of Algorithm 11. However, we have not discussed the property of the classical attribute exploration that the number of questions which the expert confirms is minimal. Indeed, we cannot expect that from our generalization, as we have not opposed any restriction on the order in which implications are asked. It is therefore possible to ask an implication $A \longrightarrow B$, which is confirmed, just to ask in the next iteration an implication $A \longrightarrow C$ with $C \supseteq B$, which might also get confirmed. It is therefore advisable to always ask implications with \subseteq -maximal conclusions. However, even in that case it might not be clear whether the number of confirmed implications asked is really minimal.

We therefore want to discuss in this section whether it is possible to modify our general attribute exploration such that the number of questions asked such that the expert confirms is the smallest possible. For this we shall try to adapt the computation of undecided implications from the classical case.

Let us recall how implications asked to a domain expert p are computed in the case of classical attribute exploration, as discussed in Algorithm 5. For this suppose that we are in a certain iteration of the algorithm, with known implications \mathcal{K} , working context \mathbb{K} and P the last computed premise. Further suppose that we have fixed a total order on the set M before the start of the algorithm, which induces a lectic order \preceq on $\mathfrak{P}(M)$. Then, in the classical case, we compute the lectically smallest set $Q \subseteq M$ after P that is closed under \mathcal{K} and that is not an intent of \mathbb{K} . The implication $Q \longrightarrow Q''$ is then asked to p .

Computing the lectically next set after a set P can be done using the Next-Closure algorithm [7]. However, for theoretical considerations we can neglect lectic orderings, as we shall see in a moment.

Let M be a finite set. To guarantee that the number of confirmed implications is as small as possible, we change step ii to:

- ii'. Let $A \subseteq M$ be such that $A = c_{\text{cert}}(A) \subsetneq c_{\text{univ}}(A)$ and A is \subseteq -minimal with respect to this property. Consider the implication $A \longrightarrow c_{\text{univ}}(A)$.

This is a generalization of the corresponding step in the classical case. If P is the premise of the last implication asked, then the lectically next set Q after P is a \subseteq -minimal set with $Q = \mathcal{K}(Q) \subsetneq Q''$, and the implication $Q \longrightarrow Q''$ is asked next.

Before we give the formal statement of the fact that this indeed yields an algorithm that always asks a minimal number of confirmed implications, we shall give the following definition.

Definition 16 Let c_1, c_2 be two closure operators on a finite set M and let $P \subseteq M$. Then P is said to be c_1 -pseudoclosed under c_2 if and only if

- i. $c_1(P) = P$,
- ii. $c_2(P) \neq P$,
- iii. for all $Q \subsetneq P$ being c_1 -pseudoclosed under c_2 it follows that $c_2(Q) \subseteq P$. ◇

Theorem 17 Consider Algorithm 11 with step ii replaced by step ii'.

Let M be a finite set, q a partial domain expert on M , c_{cert} , c_{univ} closure operators on M such that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{univ}}(\cdot)$. Let \mathcal{K} be the set of confirmed implications during the run of the algorithm with input c_{cert} , c_{univ} and q , and let c be the returned closure operator.

Then the premises of the implications in \mathcal{K} are exactly the c_{cert} -pseudoclosed sets of c .

Proof We show that a set $A \subseteq M$ is a c_{cert} -pseudoclosed set of c if and only if the implication $A \rightarrow c(A)$ is asked to and confirmed by q . We shall do so using well-founded induction, which is possible since M is finite.

Let A be a premise of a confirmed implication $A \rightarrow B$. It follows that $B = c'_{\text{univ}}(A)$ for the corresponding value of c'_{univ} in the iteration in which $A \rightarrow B$ is asked to q . Then A is closed under c_{cert} and under all currently known implications, i. e. under

$$\{ X \rightarrow Y \mid (X \rightarrow Y) \in \mathcal{K}, X \subseteq A \}.$$

Suppose that there exists an implication $(X \rightarrow Y) \in \mathcal{K}$ such that $X \subseteq B$. Then $Y \subseteq c'_{\text{univ}}(B) = c'_{\text{univ}}(A) = B$. Therefore, B is closed under \mathcal{K} and hence $B = c(A)$.

We shall show next that A is a c_{cert} -pseudoclosed set of c . We already know that A is closed under c_{cert} . Furthermore, since $A \rightarrow B$ is asked to q , $B \neq A$ and therefore $A \neq c(A)$.

Let $R \subsetneq A$ be a c_{cert} -pseudoclosed set of c . By the induction hypothesis, $R \rightarrow c(R)$ is asked to and confirmed by q . Since A is closed under all those implications, it follows that $c(R) \subseteq A$ as required.

Conversely, let A be a c_{cert} -pseudoclosed set of c . By the induction hypothesis, for all c_{cert} -pseudoclosed sets $R \subsetneq A$ the implication $R \rightarrow c(R)$ is asked to and confirmed by q . Since $c(R) \subseteq A$ and $c_{\text{cert}}(A) = A$ it follows that A is \subseteq -minimal with respect to being closed under c_{cert} and all confirmed implications $X \rightarrow Y$ with $X \subseteq A$. Therefore, $A \rightarrow c'_{\text{univ}}(A)$ will be asked in a certain iteration, with the corresponding value of c'_{univ} . Since $c(A) \subseteq c'_{\text{univ}}(A)$ and $A \neq c(A)$, after a finite number of counterexamples $A \rightarrow c(A)$ will be asked to and confirmed by q . \square

Recall the fact that the set

$$\mathcal{K} := \{ P \rightarrow c(P) \mid P \text{ is } c_{\text{cert}}\text{-pseudoclosed set of } c \}$$

has minimal cardinality such that every set $A \subseteq M$ is closed under c if and only if A is closed under c_{cert} and \mathcal{K} . This has been proven in [5] for the case of $c_{\text{cert}} = \mathcal{K}(\cdot)$ for a set $\mathcal{K} \subseteq \text{Imp}(M)$ and $c = (\cdot)''$ for some given formal context \mathbb{K} with $\mathbb{K} \models \mathcal{K}$. However, the proof given there also holds in our general setting.

Summing up, we obtain our desired result.

Corollary 18 The number of confirmed implications during the run of the general attribute exploration algorithm is as small as possible.

6 Conclusions

Starting from a classical formulation of attribute exploration using domain experts, we have presented a more general formulation of attribute exploration that is able to work with abstractly given closure operators and can handle partially given counterexamples. We have also seen that most of the properties of classical attribute exploration remain in general or, as in the case of minimality of confirmed implications, under certain restrictions.

References

- [1] Franz Baader and Felix Distel. A finite basis for the set of EL-implications holding in a finite model. In Raoul Medina and Sergei Obiedkov, editors, *Proceedings of the 6th International Conference on Formal Concept Analysis, (ICFCA 2008)*, volume 4933 of *Lecture Notes in Artificial Intelligence*, pages 46–61. Springer Verlag, 2008.
- [2] Franz Baader and Felix Distel. Exploring finite models in the description logic $\mathcal{EL}_{\text{gfp}}$. In Sébastien Ferré and Sebastian Rudolph, editors, *Proceedings of the 7th International Conference on Formal Concept Analysis, (ICFCA 2009)*, volume 5548 of *Lecture Notes in Artificial Intelligence*, pages 146–161. Springer Verlag, 2009.
- [3] Franz Baader, Bernhard Ganter, Ulrike Sattler, and Baris Sertkaya. Completing description logic knowledge bases using formal concept analysis. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 230–235. AAAI Press, 2007.
- [4] Felix Distel. Hardness of Enumerating Pseudo-intents in the Llectic Order. In Kwuida and Sertkaya [10], pages 124–137.
- [5] Felix Distel. *Learning Description Logic Knowledge Bases from Data Using Methods from Formal Concept Analysis*. PhD thesis, TU Dresden, 2011.
- [6] Bernhard Ganter. Attribute exploration with background knowledge. *Theor. Comput. Sci.*, 217(2):215–233, 1999.
- [7] Bernhard Ganter. Two basic algorithms in concept analysis. In Kwuida and Sertkaya [10], pages 312–340.
- [8] Bernhard Ganter, Sergei Obiedkov, Sebastian Rudolph, and Gerd Stumme. Conceptual Exploration. to appear.
- [9] Bernhard Ganter and Rudolph Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin-Heidelberg, 1999.
- [10] Léonard Kwuida and Baris Sertkaya, editors. *Formal Concept Analysis, 8th International Conference, ICFCA 2010, Agadir, Morocco, March 15-18, 2010. Proceedings*, volume 5986 of *Lecture Notes in Computer Science*. Springer, 2010.

- [11] Gerd Stumme. Attribute exploration with background implications and exceptions. In H.-H. Bock and W. Polasek, editors, *Data Analysis and Information Systems. Statistical and Conceptual approaches. Proc. GfKI'95. Studies in Classification, Data Analysis, and Knowledge Organization 7*, pages 457–469, Heidelberg, 1996. Springer.